We test your code the same way you should test it: by calling the functions that you wrote with various arguments and making sure that the outputs are what they are supposed to be.

Before we can call any functions in your workweek.py, Python has to load it. That means that all the statements in it will be executed, except when we load your workweek.py, __name__ will not be __main__, and so the if __name__ == __main__ block will not be executed.

That means that if you want to assign initial values to global variables, you must do so outside the if __name__ == __main__ block. This also means that you should be careful not to have any statements that change the values of global variables outside any functions and the if __name__ == __main__ block.

Here is a part of the solution to last week’s lab that would fail automatic testing:

```python
def add(to_add):
    global current_value
    current_value += to_add

if __name__ == __main__:
    current_value = 0  # BAD CODE: current_value is only initialized if
    # __name__ is __main__’ (NOTE: in itself, the code
    # is fine. It’s bad because this is the only place
    # where current_value is initialized

add(2)
display_current_value()  # expected: 2
add(5)  # BAD CODE: even if current_value were assigned the correct initial
        # value, this line would mess it up by changing the initial value to 5
    display_current_value()  # expected: 7 (this will only be 7 if __name__ is __main__)
```

Here is the corrected version:

```python
current_value = 0  # Initializing the global variable outside any if
                    # any if blocks or functions that are not
                    # guaranteed to be called

def add(to_add):
    global current_value
    current_value += to_add

if __name__ == __main__:
    current_value = 0  # Still good practice to initialize all global
                        # variables before trying each test case

add(2)  # It’s okay to change the values of global
        # variables inside functions and inside the if
        # __name__ == __main__ block
    display_current_value()  # expected: 2
add(5)
    display_current_value()  # expected: 7
```

Engineering Science, University of Toronto
Summary

• Initialize (assign the initial value to) global variables outside the `if __name__ == __main__` block

• After initializing them, only change the values of global variables inside functions and the `if __name__ == __main__` block

• Make it so that if we replace the code in your `if __name__ == __main__` block with our code, our code will work correctly.